The size and alignment of C and XC's data types are not specified by the language. This allows the size of int to be set to the natural word size of the target device, ensuring the fastest possible performance for many computations. It also allows the alignment to be set wide enough to enable efficient memory loads and stores. Figure 1 represents the size and alignment of the data types specified by the xCORE Application Binary Interface (see XM-000967-PC), which provides a standard interface for linking objects compiled from both C and XC.

Data Type	Size	Align	Supported		Meaning
	(bits)	(bits)	xc	С	-
char	8	8	1	1	Character type
short	16	16	1	1	Short integer
int	32	32	1	1	Native integer
long	32	32	1	1	Long integer
long long	64	32	X	1	Long long integer
float	32	32	×	1	32-bit IEEE float
double	64	32	X	1	64-bit IEEE float
long double	64	32	X	1	64-bit IEEE float
void *	32	32	X	1	Data pointer
port	32	32	1	x	Port
timer	32	32	1	x	Timer
chanend	32	32	1	X	Channel end

Figure 1: Size and alignment of data types on XS1 devices

In addition:

- ▶ The char type is by default unsigned.
- ▶ The types char, short and int may be specified in a bit-field's declaration.
- A zero-width bit-field forces padding until the next bit-offset aligned with the bit-field's declared type.

REV B

- ▶ The notional transfer type of a port is unsigned int (32 bits).
- ▶ The notional counter type of a port is unsigned short (16 bits).
- ▶ The notional counter type of a timer is unsigned int (32 bits).

Publication Date: 2013/11/11

XMOS © 2013, All Rights Reserved

Xmos Ltd. is the owner or licensee of the information in this document and is providing it to you "AS IS" with no warranty of any kind, express or implied and shall have no liability in relation to its use. Xmos Ltd. makes no representation that the information, or any particular implementation thereof, is or will be free from any claims of infringement and again, shall have no liability in relation to any such claims.

